Using MCMC Sampling to Calibrate a Computer Model of a Geothermal Field

Tiangang Cui
Department of Engineering Science
The University of Auckland
New Zealand
tcui001@aucklanduni.ac.nz

Colin Fox
Department of Physics
University of Otago
New Zealand
fox@physics.otago.ac.nz

Michael O'Sullivan
Department of Engineering Science
The University of Auckland
New Zealand
m.osullivan@auckland.ac.nz

Geoff Nicholls
Department of Statistics
Oxford University
UK
nicholls@stats.ox.ac.uk

Abstract

We take a Bayesian approach to the calibration of an eight-parameter model of a geothermal field, analyzing measured well-test data. The posterior distribution over parameters for each of three scenarios, using different training data subsets, is explored using Markov chain Monte Carlo sampling. A novel parallel rejection algorithm is used to reduce

computation time. Caparison across scenarios indicates model error. Comparison of one scenario with a previous least-squares estimate for the same model and data set shows that sample-based statistics give a more robust estimate than gradient-based least-squares, in less compute time.

Keywords: geothermal modelling, multiphase flow, inverse problems, Bayesian inference

1 Introduction

In many geothermal systems, a heat flux from the magma causes convection of groundwater in the overlying surface layers, with hot water rising and cool water descending (Grant, Donaldson, and Bixley 1982). The size and shape of the resulting convective plume is determined by the deep heat flux and by the shallow geological structures (typically 0-6km in depth) such as major faults, fractured zones and capping formations. When a geothermal system is exploited for energy production, wells are drilled into the hot zone of the convective plume and water and steam are extracted. To plan and manage such geothermal developments, computer models of the geothermal system are set up and used to investigate future scenarios and management options. These computer models are based on the numerical solution of the nonlinear partial differential equations which represent mass conservation, energy conservation and Darcy's law for the underground multiphase

flow. The numerical simulator TOUGH2 (Pruess 1991) is used in our study. TOUGH2 is a very general code that can simulate non-isothermal, multiphase, multicomponent flows in heterogeneous, anisotropic porous media.

At present, computer models of geothermal reservoirs are "calibrated" manually by adjusting the input parameters such as permeability, porosity and boundary conditions to fit measured data such as near-surface pressure and enthalpy. Calibrating computer models is difficult since model simulation is computationally expensive, field observations are sparse, and the rock properties are highly heterogeneous and anisotropic. These difficulties make the calibration process extremely time consuming and demanding in terms of human input, and it may not be obvious how to adjust the structure of the model to improve its fit with measured data.

Traditionally, calibration consists of making a point estimate of the parameters by best-fitting to measured data, with subsequent model prediction using that point estimate. In a previous study, the computer program ITOUGH2 (Finsterle 1993) was used to automate this process. ITOUGH2 uses a gradient-based optimization method to minimize the sum of squares of differences between measured data and TOUGH2 simulated data. ITOUGH2 has been applied to calibration of a few large-scale models of geothermal fields (Porras, Tanaka, Fujii, and Itoi 2007; Kiryukhin, Asaulova, and Finsterle 2008) and also to simple models with few parameters using measured data sets from well tests. Because of the strongly non-linear interaction

between parameters in a geothermal model, difficulties arise when the estimation involves many parameters (Mannington, O'Sullivan, and Bullivant 2004; Roylance, Yeh, and Lin 2003). Then the data-misfit function is characterized by local optima, extensive flat planes in multi-dimensional space, or is sometimes discontinuous.

Since TOUGH2 handles the integration of time implicitly, ITOUGH2 must evaluate the gradient numerically by perturbing the parameter for each dimension, with a TOUGH2 simulation required for each perturbation. Hence, ITOUGH2 is very computationally demanding and the computing time for each gradient evaluation is proportional to the dimension of the model parameter. For a 1D model with 6 parameters and about 60 seconds simulation time (Roylance, Yeh, and Lin 2003), ITOUGH2 needs several hours searching for a local minima. Roylance, Yeh, and Lin (2003) ran ITOUGH2 many times from different starting points to find an "optimal" solution that generates a reasonably good fit to data. They found that the "optimal" solution was still not able to make good predictions. To model a large scale reservoir, several thousands parameters are needed for the 3D model, requiring many hours or days of computation time. That makes ITOUGH2, or similar methodologies, impractical for calibrating large-scale models.

In this paper we take a Bayesian approach to "calibration", and explore the posterior distribution over feasible parameters conditioned on a training

data set of observed near-surface pressure and enthalpy. This then defines a predictive density over observed data; we compare predicted data to the measured data not in the training set to validate our model and calibration. We use Markov chain Monte Carlo (MCMC) sampling to explore the posterior density over parameters. Model simulation then gives samples from the predictive density for observed data.

The use of stochastic algorithms, such as MCMC, actually overcomes many of the difficulties associated with gradient-based optimization. In particular, the local minima that cause global convergence to be virtually impossible to achieve with programs such as ITOUGH2, present no difficulties to the MCMC that we use here.

Due to the high dimensionality of parameter space and the non-linearity of our data simulation step, the MCMC requires hundreds of thousands of iterations to adequately explore the posterior distribution. Since the likelihood evaluation requires a computationally demanding data simulation, the MCMC sampling is computationally very expensive. To improve the performance of sampling we introduce a novel parallel rejection scheme that parallelizes the Metropolis-Hastings algorithm. This is achieved by parallelizing the target density evaluation for states that are rejected. This scheme has the benefit that it was not necessary to modify the legacy Fortran code TOUGH2 and can be implemented on a cluster of PCs or workstations.

To test the method we analyze pressure and enthalpy data from a geother-

mal well measured during a discharge test. Traditionally, these data are used to estimate the permeability of the underlying geothermal field. In our study, we aim to sample from the posterior distribution over parameters conditioned on these data and construct a predictive density to forecast the pressure and enthalpy changes. For the computer model we build a single layer 1D radially symmetric model to simulate the pressure and enthalpy response. There are total of 8 associated model parameters, which represent the unknown permeability, porosity and unknown initial conditions. We also compare our results to an estimate given by running ITOUGH2 using the same data set (Finsterle, Pruess, Bullivant, and O'Sullivan 1997).

The computer model of the geothermal reservoir is introduced in section 2. The observed data and the likelihood function are discussed in section 3. Section 4 defines the prior distribution over model parameters. The proposal distribution used in the Metropolis-Hastings algorithm and the parallel rejection scheme are introduced in section 5. Output statistics obtained by sample-based inference are shown in section 6, and compared with the gradient-based optimization result obtained by Finsterle, Pruess, Bullivant, and O'Sullivan (1997).

2 FORWARD MAP

Multiphase non-isothermal flow in a geothermal reservoir can be simulated using the TOUGH2 code (Pruess 1991). In TOUGH2, two phase flow (wa-

ter and vapour) is modelled by general mass balance and energy balance equations. For an arbitrary subdomain Ω with bounding surface $\partial\Omega$, the balance equations can be written in the form

$$\frac{d}{dt}\int_{\Omega} M\, dV = \int_{\partial\Omega} Q \cdot \hat{n}\, d\Gamma + \int_{\Omega} q\, dV. \tag{1}$$

The accumulation term M represents mass (M_{m}) or energy (M_{e}) per unit volume, defined by

$$\begin{aligned} M_{\mathrm{m}} &= \phi\,(\rho_{\mathrm{l}} S_{\mathrm{l}} + \rho_{\mathrm{v}} S_{\mathrm{v}}), \\ M_{\mathrm{e}} &= (1-\phi)\,\rho_{\mathrm{r}} c_{\mathrm{r}} T + \phi\,(\rho_{\mathrm{l}} u_{\mathrm{l}} S_{\mathrm{l}} + \rho_{\mathrm{v}} u_{\mathrm{l}} S_{\mathrm{v}}). \end{aligned} \tag{2}$$

Here ϕ is porosity, ρ is density, S is saturation, u is specific internal energy, c is specific heat and T is the temperature. The subscripts l, v and r indicate the quantities pertaining to liquid, vapour and rock, respectively. Note that: $S_{\mathrm{l}} + S_{\mathrm{v}} = 1$.

The mass flux term is given as a sum over liquid and vapour phase:

$$Q_{\mathrm{m}} = \sum_{\beta=\mathrm{l,v}} \frac{k k_{\mathrm{r}\beta}}{\nu_{\beta}} (\nabla p - \rho_{\beta} \underline{g}). \tag{3}$$

Here k is a diagonal second order permeability tensor in 3-dimensions, $\underline{g}$ is the acceleration due to gravity, ∇p is the pressure gradient acting on the fluid flow, and ν is the kinematic viscosities. Relative permeability $k_{r\beta}$ is

introduced to account for the interference between liquid and vapour phases as they move through the rock matrix in the geothermal reservoir. There are several empirically derived curves available to model $k_{r\beta}$ as functions of vapour saturation S_v. We use the van Genuchten-Mualem model (van Genuchten 1980) with $S_{rv} = 0$, defined by:

$$k_{rl} = \begin{cases} \sqrt{S^*}\left\{1 - \left(1 - [S^*]^{1/m}\right)^m\right\}^2 & \text{if } S_l < S_{ls} \\ 1 & \text{if } S_l \geq S_{ls}, \end{cases}$$

where

$$\begin{aligned} S^* &= (S_l - S_{rl})/(S_{ls} - S_{rl}); \\ k_{rv} &= \begin{cases} 1 - k_{rl} & \text{if } S_{rv} = 0 \\ \left(1 - \hat{S}\right)^2 \left(1 - \hat{S}^2\right) & \text{if } S_{rv} > 0, \end{cases} \\ \hat{S} &= (S_l - S_{rl})/(1 - S_{rl} - S_{rv}) \end{aligned} \tag{4}$$

subject to the restrictions

$$\begin{aligned} &0 \leq k_{rl}, k_{rv} \leq 1, \\ &S_{rl} + S_{rv} < 1, \\ &S_{rl} < S_{ls}. \end{aligned} \tag{5}$$

Except for the hyperparameters m, S_{rl}, S_{ls} and S_{rv}, all the remaining parameters in the van Genuchten-Mualem model are components of the state of the geothermal system.

Energy is carried by the movement of steam and water, and by thermal conduction. Hence the energy flux is given by

$$Q_{\mathrm{e}} = \sum_{\beta=\mathrm{l,v}} \frac{k k_{\mathrm{r}\beta}}{\nu_\beta} (\nabla p - \rho_\beta \underline{g}) h_\beta - K \nabla T. \tag{6}$$

Here h denotes specific enthalpy and K is the thermal conductivity in a saturated medium.

The accumulation term q in equation 1 represents the mass and heat sources or sinks in region Ω. The thermolphysical properties of liquid water and steam (such as density, viscosity, specific internal energy, specific heat and specific enthalpy) are calculated using steam table equations given by the International Formulation Committee (1967).

Spatial discretization of equations 1 is based on an integrated finite difference or finite volume method, which is implemented in the existing Fortran code TOUGH2. To guarantee the stability of simulation, TOUGH2 uses a fully implicit method for numerical integration in time, and upstream weighting for calculating the velocity of fluid movement between adjacent blocks (in equations 3 and 6). For each time step, the Newton-Raphson method is used to solve the resulting system of non-linear difference equations. TOUGH2 uses a preconditioned conjugate gradient sparse matrix solver for solving the linear equations at each Newton-Raphson iteration (see Pruess 1991).

As mentioned above, our inference is based on samples drawn from the posterior distribution over field parameters conditioned on pressure data and flowing enthalpy data measured at the well head during a well discharge test. We model the geothermal field as a single layer radially symmetric model with homogeneous rock properties. The model is located at a depth of about 1600 meters which corresponds to the feed zone of the well. Figure 1 shows a wedge of the computational grid of the model. The well is located in the central block shown at the vertex of the wedge, 10 small blocks are used adjacent to the well, followed by 65 blocks with a thickness expansion factor of 1.2. Because the volume of each outer block is large the temperature and

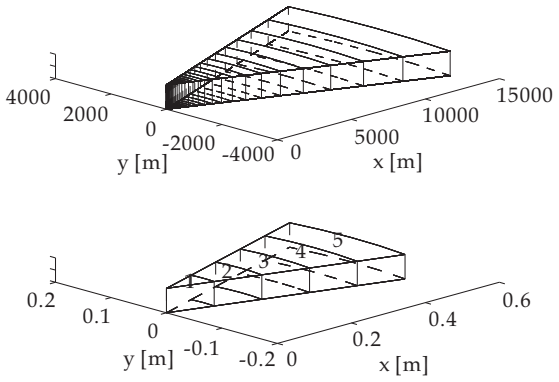


Figure 1: Finite volume grid for the well discharge test model. The top figure gives the overview of (a wedge from) the single-layered one-dimensional radially-symmetric model. The bottom figure shows the first five blocks; the well located in the first block.

pressure in those blocks do not change during the simulation, and hence can be viewed as providing a model with effectively infinite extent. On a 3.0 GHz Pentium 4 processor, roughly 20 seconds is required to simulate the

system response for 100 days of operation.

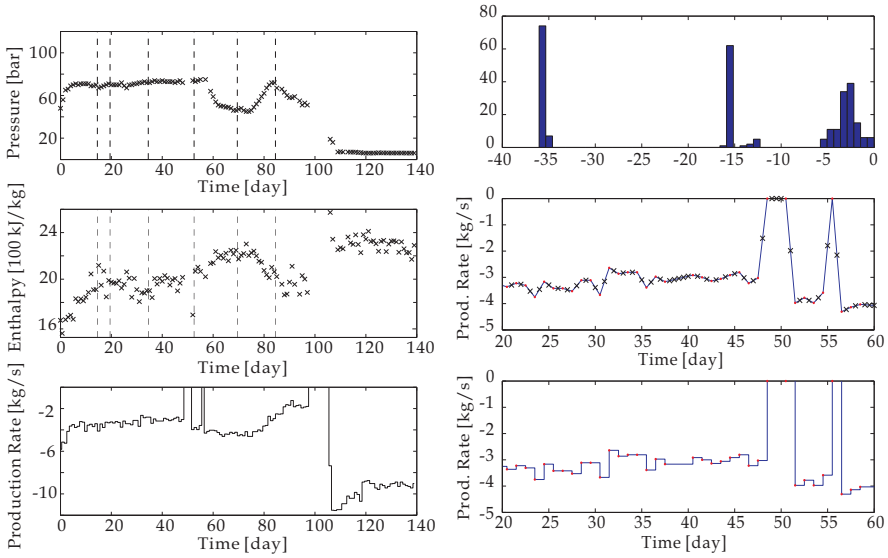
The numerical simulation generates 'down hole' values whereas observed data are measured at the well head. While the energy loss and enthalpy change along the well bore are not large, the pressure drop is significant. Accordingly, we introduce a pressure shift parameter variable p_{s} to model this difference. Since the model is assumed to be homogeneous, there is only one porosity value and one permeability value in the radial direction that need to be estimated. The initial vapour saturation ($S_{\mathrm{v}0}$) and initial pressure (p_0) are used to represent the initial thermodynamics state of the two phase system, but are not known in advance. Together with the parameters from the relative permeability model, these make up the eight unknown parameters for the data simulation:

$$\theta = (\phi, \log_{10}(k), p_0, S_{\mathrm{v}0}, p_{\mathrm{s}}, m, S_{\mathrm{rl}}, S_{\mathrm{ls}})\,. \tag{7}$$

Note that the permeability k is represented on a base 10 logarithmic scale.

3 OBSERVED DATA AND LIKELIHOOD FUNCTION

We analyze a measured data set in which production rate, flowing enthalpy, and wellhead pressure data were observed during a period of 140 days. The data is shown in figure 2(a). The well was shut down three times (see figure



(a) (b)

Figure 2: (a) Data used for the well discharge test analysis. The top, middle and bottom plot corresponding to the pressure, the flowing enthalpy and the production rate, respectively. (b) Production rate from day 20 to day 60. Top: histogram of second differences. Middle: traditional linear interpolation between data points; interpolated data points (see text) are shown by crosses. Bottom: interpolation of measured production rate used in this study.

2(a)), for 2 days around day 50, one day at day 55, and for 5 days around day 100. After day 105, the production rate was increased from about (minus) 4 kg/s to about (minus) 10 kg/s.

In our TOUGH2 model, the simulation could not continue with a low permeability value ($k < 1$ millidarcy) after day 105 because the model pressure dropped below atmospheric pressure which is not permitted in TOUGH2. As shown later, to match the data before day 105, the feasible range of permeability is usually below 1 millidarcy (i.e. $\log_{10}(k) < -15$). Figure 2(a) shows that the production rate drops between day 85 to day 95, while the

pressure drops in this period as well. This effect may be caused by interaction with other wells, which is not included in our 1D model. Further, TOUGH2 is not able to simulate the steep rise in the enthalpy data and the increase in the pressure data from day 0 to day 10. Thus, our analysis is based on the data collected from day 14 to day 84.

The production rate q_{m} is also recorded during the well discharge test at 24 hour intervals. The rate change within this interval is not specified. We found that the production rate at some points had been linearly interpolated from the adjacent measured data points. The top plot of figure 2(b) is the histogram of the second difference of production rate data with respect to time, $\triangle^2 q_{\mathrm{m}}$ shown on a base 10 logarithmic scale. Data points with second difference less than 10^{-10} are certainly the product of linear interpolation of neighbouring measured points, hence contribute no extra information. The middle plot of figure 2(b) shows the original production rate (from day 20 to day 60), the crosses are the data points with a small second difference ($< 10^{-10}$). These points are discarded in our analysis, so do not contribute to the likelihood.

The middle plot in figure 2(b) also shows the traditional *linear* interpolation of production rate between measured data points. In a previous study, Finsterle, Pruess, Bullivant, and O'Sullivan (1997) analyzed the same data set using a linearly interpolated production rate; the results are shown later in figure 11. From those plots we observe a sudden pressure and enthalpy

change around day 55 resulting from the shutting down of the well around day 55 (see figure 2(b)). In our analysis we found that when the permeability is low, the use of linear interpolation of production rate may shift down the enthalpy and bring the pressure up after day 55. This effect can be overcome by an alternative interpolation which treats the data points (bottom plot of figure 2(b)) as the value of a step function. That interpolation is consistent with the interpretation that production rate is *set* for each 24 hours rather than being a measurement of a smoothly varying quantity.

We denote the observed data as $\underline{\tilde{y}} = \left(\underline{\tilde{h}}, \underline{\tilde{p}}\right)$, where $\underline{\tilde{h}}$ and $\underline{\tilde{p}}$ are observed flowing enthalpy and well head pressure, respectively. Conditional on the unknown parameter θ, we specify a Gaussian measurement error of the form:

$$\begin{aligned} L(\underline{\tilde{y}}|\theta) \propto \exp\left\{-\frac{1}{2}\left[(\underline{\tilde{h}} - \underline{h}(\theta))'\Sigma_{\mathrm{h}}^{-1}(\underline{\tilde{h}} - \underline{h}(\theta))\right]\right\} \\ \times \exp\left\{-\frac{1}{2}\left[(\underline{\tilde{p}} - \underline{p}(\theta))'\Sigma_{\mathrm{p}}^{-1}(\underline{\tilde{p}} - \underline{p}(\theta))\right]\right\}, \end{aligned} \tag{8}$$

where $\underline{h}(\theta)$ and $\underline{p}(\theta)$ are the model output for parameter θ. The covariances of flowing enthalpy Σ_{h} and covariances of well head pressure Σ_{p} are fixed constant. Because the data observations are fairly even spaced, we assume that $\Sigma_{\mathrm{h}} = \sigma_{\mathrm{h}}^2 I_n$ and $\Sigma_{\mathrm{p}} = \sigma_{\mathrm{p}}^2 I_n$. The actual choice of standard deviations σ_{h} and σ_{p} are estimated from the residual of a smoothed spline interpolation of the time series data $\underline{\tilde{h}}$ and $\underline{\tilde{p}}$.

We divide the observed data from day 14 to day 84 into 5 sets (see figure

2(a)). We perform two analyses using different partitions of observations for training with the remaining observations used for validation. Also, an analysis with all the observations used for training is conducted to compare with the results of the previous study by Finsterle, Pruess, Bullivant, and O'Sullivan (1997). Table 1 shows the three scenarios we analyzed and the observations used for each scenario.

Scenario	Training Data [day]	σ_{h} [kJ/kg]	σ_{p} [bar]
S1	{14-19, 35-52}	50	3
S2	{14-19, 53-69}	50	3
S3	{14-84}	100	6

Table 1: The training data subsets for each of the three scenarios used to analyze the well discharge test. Columns labelled σ_{h} and σ_{p} give the estimated standard deviations of flowing enthalpy and well head pressure, respectively.

4 PRIOR DISTRIBUTION

We used expert knowledge and physical constraints to define bounds on the allowable range of each component of the parameter θ, and within that defined weak subjective prior densities. For example, the value of porosity is necessarily between 0 and 1, and commonly less than 0.5. To formulate such weak prior information into a functional form, we use a modified exponential distribution of the form

$$\pi(x) \propto \frac{x}{s}\left(1-\frac{x}{s}\right)\exp\left(\iota\frac{x}{s}\right). \tag{9}$$

Here s scales the parameters into the range $(0, 1)$ and ι is a non-zero constant which control the skewness of the distribution, setting the region of the mode. The density for various choices of ι are shown in figure 3.

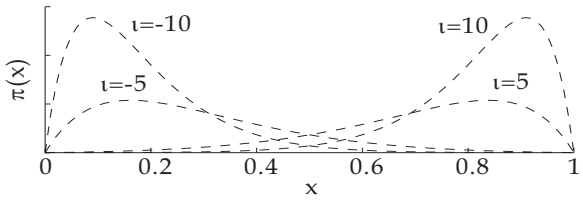


Figure 3: Plot of $\pi(x)$ for different choice of ι, where $\iota = -10, -5, 5, 10$.

The physical constraints and bounds that we used for each component of the parameter θ, and the different choices of ι for each component, are given in table 2, except the permeability k. Note that the constraints in

	ϕ[-]	m [-]	S_{rl} [-]	S_{ls} [-]	S_{v0} [-]	p_0 [bar]	p_{s} [bar]
U.B.	1	1	1	1	1	250	150
L.B.	0	0	0	0	0	50	50
ι	-5	5	2	10	-10	-2	-2

Table 2: The bounds for different components of parameter θ and the different choice of skewness parameter ι.

the van Genuchten-Mualem model (equation 5) is also included in the prior distribution. We make the prior density over permeability invariant with respect to changes of unit by specifying a uniform density over a logarithmic scale (Jaynes 1968). It is not necessary to truncate the range of allowable permeability values to make the prior proper as the range is effectively constrained by the likelihood function.

5 POSTERIOR EXPLORATION

We explore the posterior distribution

$$\pi(\theta|\tilde{d}) \propto L(\tilde{d}|\theta)\pi_{\rm pr}(\theta)$$

by MCMC sampling using Metropolis-Hastings (MH) dynamics (Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller 1953; Hastings 1970) with random walk proposals tuned to give good performance. We find it convenient to use the reversible jump formalism (Green 1995) to calculate acceptance probabilities even though the dimensionality in our problem is fixed.

5.1 Transition Kernel

To propose a new candidate parameter θ', we build L different proposal distributions $q^{(i)}(\theta'|\theta)$, $i = 1, \ldots, L$. Let $K_{q^{(i)}}(\theta', \theta)$ be the MH transition kernels with $q^{(i)}(\theta'|\theta)$ as a proposal, and let ζ_i, $i = 1, \ldots, L$, be the probability of choosing kernel i. Then the overall transition kernel is given by

$$K(\theta', \theta) = \sum_{i=1}^{L} \zeta_i K_{q^{(i)}}(\theta'|\theta). \tag{10}$$

We find the following $L = 3$ proposals are are sufficient to give good mixing and ensure ergodicity of the Markov chain over useful time scales. Within each proposal we select a subset θ_I of components of the parameter θ to update, where I represents a set of indices. The proposals are:

Translate: Translate θ_I by $\lambda \sim U(-\epsilon, \epsilon)$: $\theta'_I = \theta_I + \lambda$.

Scale: Scale θ_I by $\lambda \sim U(1/\epsilon, \epsilon)$: $\theta'_I = \lambda\theta_I$.

Shoot: Draw a two dimensional random vector $\lambda \sim U(-\epsilon_1, \epsilon_1) \times U(-\epsilon_2, \epsilon_2)$, then update the two dimensional vector θ_I by

$$\theta'_I = \theta_I + \begin{pmatrix} \cos(\gamma) & \sin(\gamma) \\ -\sin(\gamma) & \cos(\gamma) \end{pmatrix} \lambda.$$

Here γ is a given angle.

The translation proposal operates on one randomly selected component from $\{\log_{10}(k), S_{\mathrm{rl}}, \phi\}$ selected with probability $\{0.4, 0.4, 0.2\}$. The scaling proposal randomly selects one parameter from $\{\phi, m\}$ with probability $\{0.473, 0.526\}$. The shooting proposal updates a two dimensional subset of parameters selected uniformly at random from $\{\{p_0, p_{\mathrm{s}}\}, \{S_{\mathrm{ls}}, S_{\mathrm{v0}}\}\}$. The Jacobian determinant in the MH algorithm for translate and shoot proposals is 1, and is λ^{-2} for the scale proposal. We select one of the kernels with probability $\zeta = \{0.45, 0.19, 0.36\}$. These three kernels are sufficient to ensure the resulting Markov chain is aperiodic and irreducible, and hence standard results guarantee that the resulting reversible chain has the desired ergodic properties.

5.2 Parallel Rejection Algorithm

We introduce an algorithm to decrease the CPU time per MCMC update by a straightforward parallelizing of the serial MH algorithm. Consider the serial Markov chain $\{X_i\}_{i=0}^{\infty}$ at state $X_i = x_i$ for some i. Suppose that n processors are available, that we take to be independent for computing purposes. We run on each processor an independent instance of the MH algorithm initialized at state x_i to give the n independent Markov chains $\{Y_{r,k}\}_{k=0}^{\infty}$ for $r = 1, 2, \ldots, n$ with $Y_{r,0} = x_i$. We enumerate the resulting states by $t(r,k) = r + n(k-1)$ for $r = 1, 2, \ldots, n$ and $k = 1, 2, \ldots$ giving the total ordering $t = 1, 2, \ldots$. Note that t is an invertible function so we may refer to states on the parallel computer by the total ordering t, i.e. as $\{Y_t\}_{t=0}^{\infty}$. We run the n parallel chains until the first non-trivial acceptance (in the ordering t) occurring at $t_{\min}$, i.e. the minimum t for which $Y_t \neq x_i$. Then set $X_j = x_i$ for $j = i+1, i+2, \ldots, i+t_{\min}-1$ and $X_{i+t_{\min}} = Y_{t_{\min}}$, and reinitialize the n parallel chains. (The same idea has been introduced previously by Sohn (1995) in constructing a synchronous parallel version of simulated annealing.)

Algorithm 1 Parallel Rejection Algorithm

1: Initiate n parallel chains $\{Y_{r,k}\}_{k=0}^{\infty}$ for $r = 1, 2, \ldots, n$ with $Y_{r,0} = x_i$
2: Run until there is some $t_{\min}$ with $Y_{t_{\min}} \neq x_i$ AND $Y_t = x_i \ \forall t < t_{\min}$ in the ordering $t(r,k) = r + n(k-1)$
3: Set $X_j = x_i$ for $j = i+1, i+2, \ldots, i+t_{\min}-1$ and $X_{i+t_{\min}} = Y_{t_{\min}}$

In practice, each of the parallel chains can be halted after it has an

acceptance, or when it is performing an iteration with index t greater than an acceptance in another chain.

In the simplest case where time per MCMC step is constant, the speedup is achieved because the serial chain is advanced m steps in time proportional to $\lfloor m/n \rfloor + 1$ rather than time proportional to m. To estimate the performance of this algorithm, we assume each accept/reject occurs independently with probability α of acceptance. For a fixed compute time per process the speed-up factor is

$$\frac{t_{\mathrm{s}}}{t_{\mathrm{p}}} = \frac{1}{\alpha}(1 - (1-\alpha)^{n-1}) \tag{11}$$

Where t_s and t_p are the time per update for serial and parallel Markov chain, respectively.

Since the transaction time (time spent on farming out slave processes) is linearly increased as the process number increases, we need to consider that time as well. Let t_{t} and t_{F} denote the transaction time and the time spent on solving the forward problem, respectively. Then the speed-up factor is multiplied by $(t_{\mathrm{t}} + t_{\mathrm{F}})/(nt_{\mathrm{t}} + t_{\mathrm{F}})$. Let $t_{\mathrm{t}}/t_{\mathrm{F}} = \rho$ denote the faction of transaction time compare to the time of calculating forward map, then the speed-up factor with transaction time can be written as

$$\frac{t_{\mathrm{s}}}{t_{\mathrm{p}}} = \frac{(1 - (1-\alpha)^{n-1})}{\alpha} \frac{1+\rho}{1+n\rho}. \tag{12}$$

Figure 4(a) shows the theoretical speed-up factor for $\rho = 0.2$ (this is

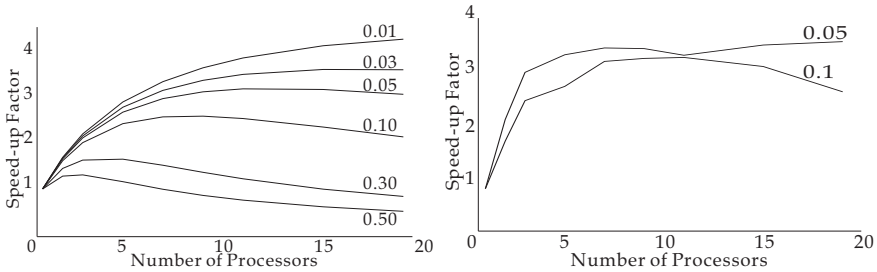


Figure 4: The speed-up factor for the parallel rejection scheme. (a) The theoretical speed-up factor. The average acceptance rates being tested here are $0.01, 0.03, 0.05, 0.1, 0.3, 0.5$ with $\rho = 0.2$. (b) The measured speed-up factor found on the computer used. The average acceptance rates are 0.05 and 0.10.

roughly the value we measured on the computer we used), $n = 1, 2, \ldots, 19$, $\alpha = \{0.01,\ 0.03,\ 0.05,\ 0.1,\ 0.3,\ 0.5\}$. Note that performance will drop when the acceptance rate is high and number of processors is large.

We implemented the parallel rejection scheme on a cluster, in which each machine consisted of several CPUs (2 or 4). The parallelization was achieved by a Perl script based on the Secure Shell. The scalable parallel random number generators library (SPRNG1.0, Mascagni and Srinivasan 2000) was used for random number generation. All the nodes on our cluster shared the same hard disk which had timing implications since TOUGH2 requires disk IO on each time step (to record the thermodynamics information). Hence we could not utilize all the computing resource as a large number of TOUGH2 process running at the same time compete with each other for disk resources. We measured performance by using up to 19 CPUs with proposal distributions set to give an acceptance rate of 5% and 10% ($\pm 1\%$). Figure

4(b) shows that the measured speed-ups are consistent with our theoretical prediction, and is slightly higher than theoretical prediction when $\alpha = 0.1$.

6 Computed Results

We ran the MCMC simulation for about 60,000 iterations for each of the scenarios given in section 3. The computing time for scenario S1, S2 and S3 were roughly 20 hours, 60 hours and 90 hours, respectively. Figure 5, 7 and 9 shows summary statistics of predicted well head pressure and flowing enthalpy for each scenario. The predictive density at each time looks Gaussian, hence we calculate sample means and standard derivations to summarize the prediction. The posterior means are shown as a solid line, while the posterior uncertainty is shown by the shaded region with width ± 3 standard deviations.

The squares in figure 5, 7 and 9 indicate data used for training, while triangles are data used for validation. Scenario S1 produces a larger error band in both pressure and enthalpy predictions after day 52, and both predictions are flatter than the observed data. The maximum mismatch in pressure and enthalpy are about 20 bars (too high) and 200 kJ/kg (too low), respectively. Scenario S2 used the data between day 53 and day 69 for training which has larger variation; Correspondingly S2 produces a much tighter predictive interval that covers the observed data better than S1. Since all the observed data points are used for training in scenario S3, that scenario effectively

implements model-based smoothing and provides an even tighter prediction interval than S2 with the model output uncertainty intervals covering the observed data reasonably well.

Figure 6, 8 and 10 show the marginal distributions for each component of θ, and show that there exists strong negative correlation between initial vapour saturation (S_{v0}) and S_{ls}. In the scenario S1, there is some negative correlation between $\log_{10}(k)$ and p_0 and strong negative correlation between p_0 and p_{s}. In the scenario S2, both pairs $\log_{10}(k), p_0$ and p_0, p_{s} are strongly correlated, whereas $\log_{10}(k)$, p_0 and p_{s} are negatively correlated in scenario S3.

The summary statistics for these three scenarios show that the mean prediction and uncertainty intervals are different when different training data sets are used. Together with the plots of the marginal distributions for each component of θ (figure 6, 8 and 10), we can conclude that the posterior distribution conditioned on the different training data sets have different shapes, and modes in different regions of the parameter space, even though the data sets are measured from the same physical process. That clearly indicates that there is appreciable modelling error, i.e., that there is a discrepancy between the computer model used for simulation and the physical process being measured.

We also compare our analysis with the previous study by Finsterle, Pruess, Bullivant, and O'Sullivan (1997) who calibrated the same TOUGH2

model based on the the same observed data from day 10 to day 85 (our scenario S3) using gradient-based optimization implemented in ITOUGH2; their result is shown in figure 11. Our scenarios S2 and S3 produce better mean estimation of both pressure and enthalpy output, and provide a more accurate uncertainty interval that is based on the posterior distribution rather than an *ad hoc* Gaussian assumption. The estimated parameter of Finsterle, Pruess, Bullivant, and O'Sullivan (1997) is compared with our posterior distribution of scenario S3 in the figure 10, the red lines in the histograms show their least-squares estimate of parameter values. We can observe that the least-squares estimates of permeability $\log_{10}(k)$, the van Genuchten parameter m, and the constant pressure shift p_{s} all lie out of the significant support of their marginal distributions. This gives evidence that the gradient based optimization has converged to a local minimum, which in this case lies in a region of very low posterior probability.

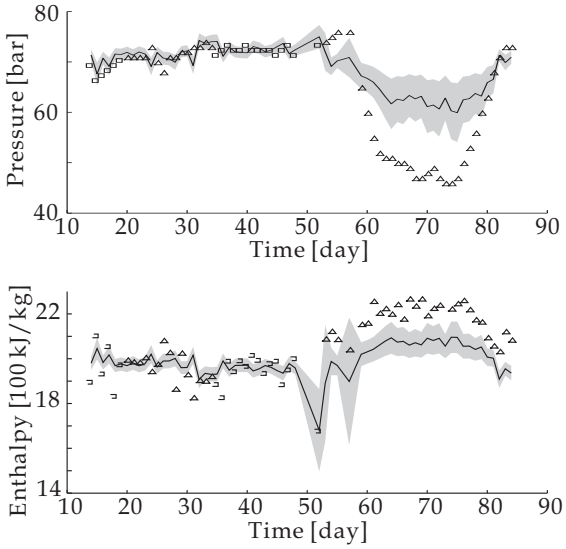


Figure 5: Results for scenario S1. From top to bottom: pressure and flowing enthalpy. The black lines are mean prediction and the gray bands are uncertainty intervals. The squares and triangles are training data points and validating data points, respectively.

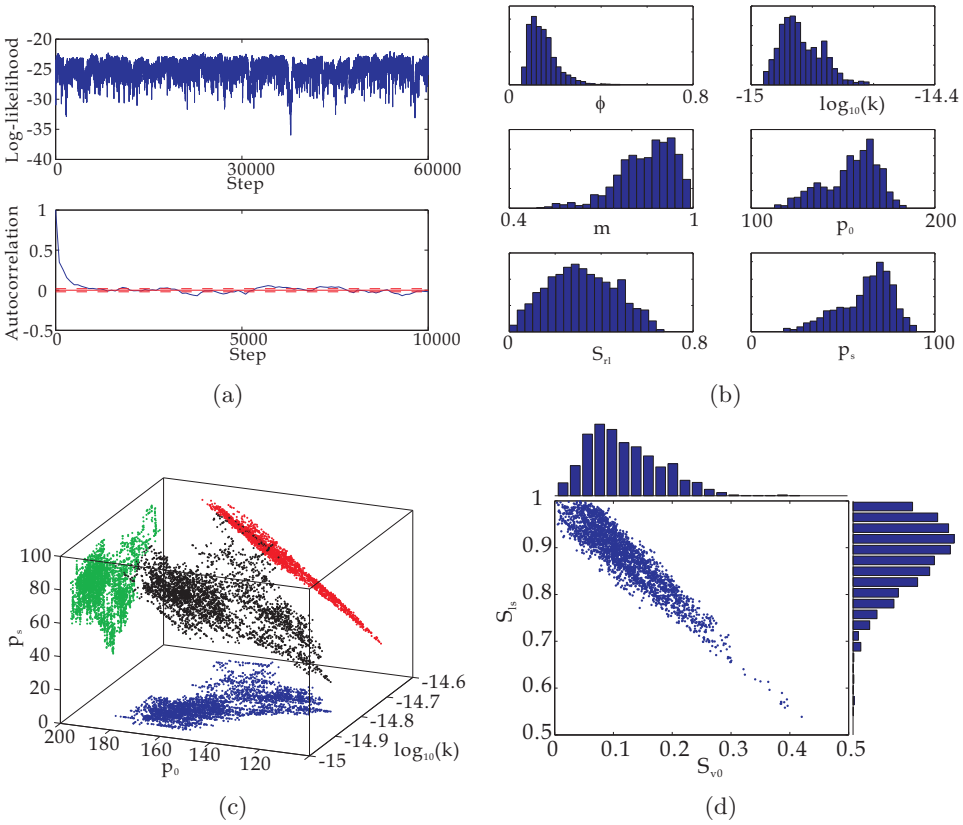


Figure 6: Output analysis for scenario S1. (a) Trace of log-likelihood (top) and autocorrelation of log-likelihood (bottom). (b) Histogram of the marginal distribution for ϕ, m, S_{rl}, $\log_{10}(k)$, p_0 and p_s. (c) Scatter plot of the joint marginal distribution over $\log_{10}(k)$, p_0 and p_s. (d) Histogram for the marginal distribution of S_{v0} and S_{ls} and scatter plot of their joint marginal distribution.

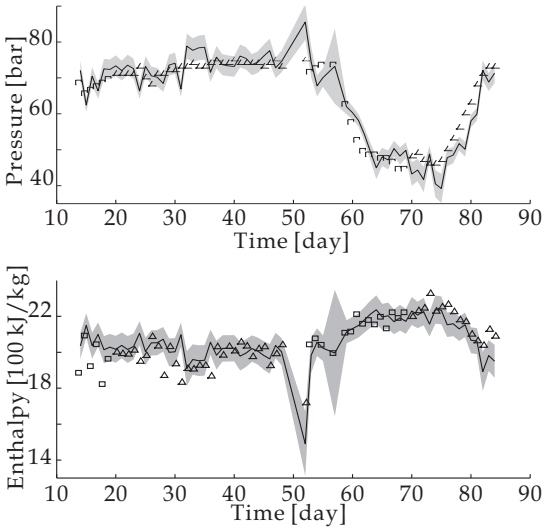


Figure 7: Results for scenario S2, with same plots as figure 5.

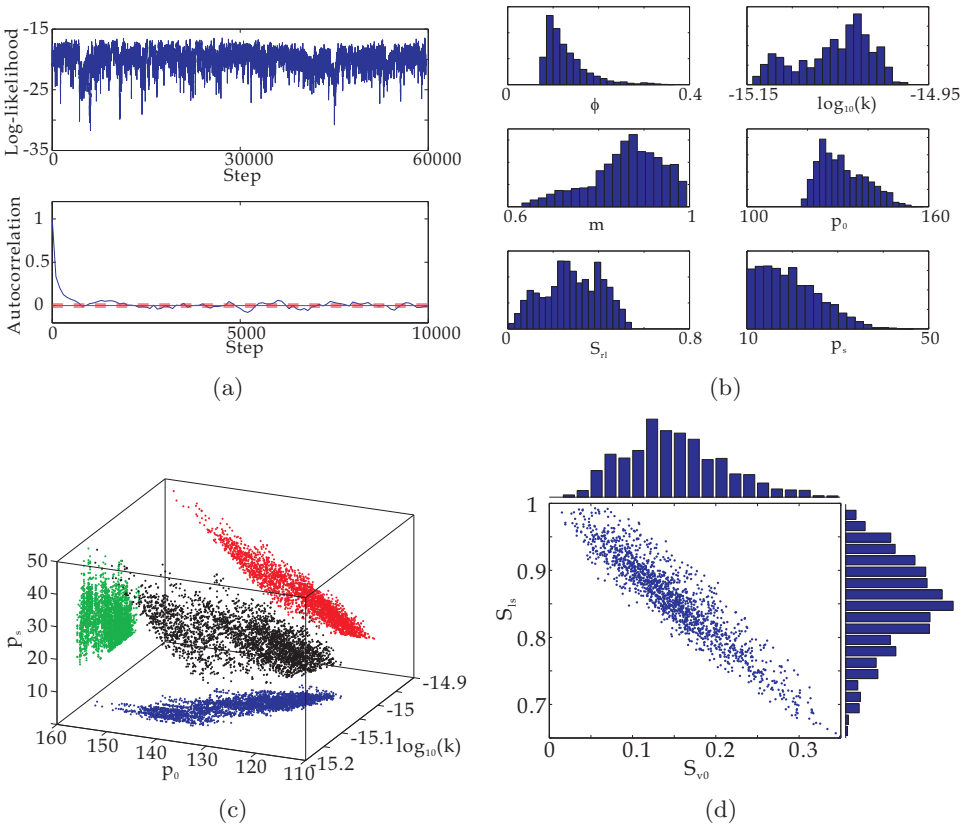


Figure 8: Output analysis for scenario S2, with subplots as figure 6.

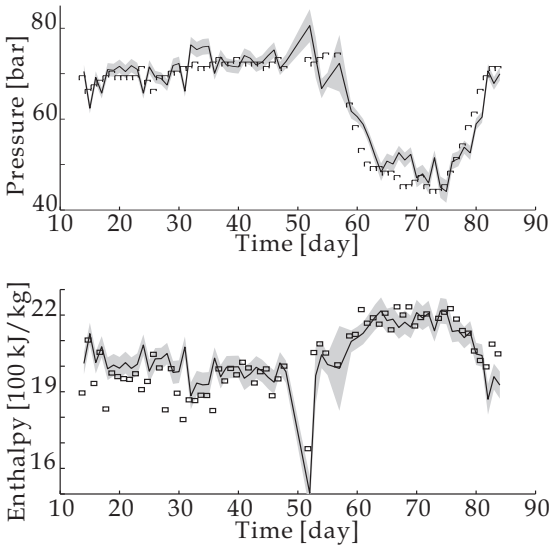


Figure 9: Results for scenario S3, with same plots as figure 5.

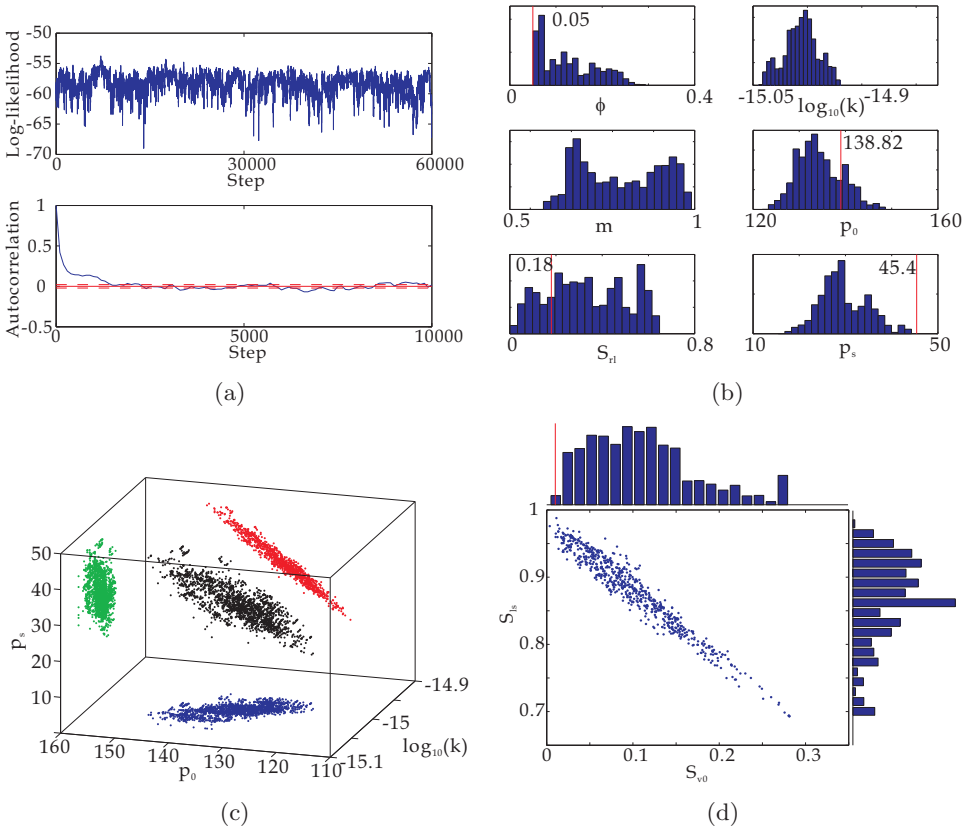


Figure 10: Output analysis for scenario S3, with subplots as figure 6. The red lines in the histograms in (b) and (c) mark the least-squares estimates reported by Finsterle, Pruess, Bullivant, and O'Sullivan (1997). The least-squares estimates they reported for permeability $\log_{10}(k)$was -14.48, and for m was 0.408; these estimates lie off the ordinate range of the histograms. They did not report the least-squares estimate for S_{ls}.

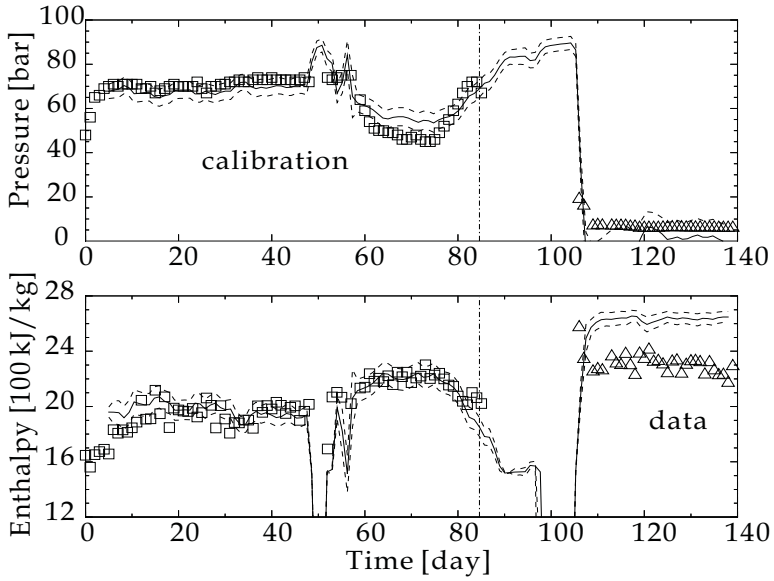


Figure 11: Results of the analysis on the same problem (as our S3) obtained by Finsterle, Pruess, Bullivant, and O'Sullivan (1997). The squares and triangles are observed data used for calibration and validation, respectively. The the solid lines are the the best fit obtained by ITOUGH2, and the regions bounded by the dashed lines are the uncertainty interval.

7 Discussion

In this paper we used a practical sample-based method to calibrate a simple model of a geothermal field. The parallel rejection version of the Metropolis-Hastings algorithm was introduced to decrease the CPU time per acceptance in MCMC sampling.

To make model predictions and to be able to quantify the uncertainty in prediction, the posterior distribution over the model parameters conditioned on observed data is used in our study. In section 6 our predictive density shows a more robust and accurate result than the traditional least-squares estimate, and the MCMC sampling is not limited by the constraints associated with gradient based optimization such as local minima. As discussed in the introduction, ITOUGH2 users need to restart the search many times to find the global optima and the gradient can only be numerically evaluated. Overall, we found that ITOUGH2 is computationally slower than MCMC in this study.

By comparing the three scenarios that we analyzed we find that the inversion result is significantly improved when data collected between day 53 and day 69 are used. This set of data corresponds to an increased production rate during the same period, the production rate (figure 2(b)) is increased from about (minus) 3 kg/s to more than (minus) 4 kg/s after day 52. If all the training data is from one regime (in scenario S1, all the data are measured with low production rate), the model could not reproduce the

system behavior across different regimes, i.e., give predictions after day 53. When data sets from both regimes are used (as in scenario S2) the solution behaves more like an average through the different regimes, hence it could follow the trend of observed data well. This suggests that including training data from different regimes is more important than the amount of data used in training. This behaviour also indicates that our numerical model has errors.

We have used a simple stationary model for measurement errors since there is no indication that data collected at different times have different error process or different noise level. One possible reason is that the modelling process contributes a different source of error into our inference. When modelling error have a different structure to the Gaussian white noise in the measurement process we assumed, the posterior distribution behaves quite differently in different regime. This fact is also supported from another aspect, different training data results posterior distribution drifting and changing the shape (figure 6, 8 and 10).

We have demonstrated that sample-based inference using MCMC sampling is a potential means of calibrating numerical models of geothermal fields in a case with few (tens of) parameters. The MCMC presented here outperformed gradient-based optimization both in terms of quality of calibration and, perhaps surprisingly, in computational speed. We are currently extending this investigation to examples that require many thousands of

parameters. In those cases gradient-based optimization is known to be ineffectual, while MCMC based calibration looks promising, though computationally demanding.

References

Finsterle, S. (1993), *ITOUGH2 User's Guide*, Lawrence Berkeley National Laboratory, Berkeley, California.

Finsterle, S., Pruess, K., Bullivant, D., and O'Sullivan, M. J. (1997), "Application of inverse modeling to geothermal reservoir simulation," in *Twenty-Second Workshop on Geothermal Reservoir Engineering*, Stanford University, Stanford, California, pp. 309–316.

Grant, M. A., Donaldson, I. G., and Bixley, P. F. (1982), *Geothermal Reservoir Engineering*, Academic Press, New York.

Green, P. J. (1995), "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, 82, 711–732.

Hastings, W. (1970), "Monte Carlo sampling using Markov chains and their applications," *Biometrika*, 57, 97–109.

International Formulation Committee (1967), *A Formulation of the Thermodynamic Properties of Ordinary Water Substance*, IFC Secretariat.

Jaynes, E. T. (1968), "Prior Probability," *IEEE Transactions On System Science and Cybernetics*, sec-4 (3), 227–241.

Kiryukhin, A. V., Asaulova, N. P., and Finsterle, S. (2008), "Inverse modeling and forecasting for the exploitation of the Pauzhetsky geothermal field, Kamchatka, Russia." *Geothermics*, 37, 540–562.

Mannington, W., O'Sullivan, M. J., and Bullivant, D. (2004), "Computer modelling of the Wairakei-Tauhara geothermal system, New Zealand," *Geothermics*, 33, 401–419.

Mascagni, M. and Srinivasan, A. (2000), "Algorithm 806: SPRNG: A Scalable Library for Pseudorandom Number Generation," *ACM Transactions on Mathematical Software*, 26, 436–461.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equation of state calculations by fast computing machines," *Journal of chemical physics*, 21, 1087–1092.

Porras, E. A., Tanaka, T., Fujii, H., and Itoi, R. (2007), "Numerical modeling of the Momotombo geothermal system, Nicaragua." *Geothermics*, 36, 304–329.

Pruess, K. (1991), *TOUGH2 - A General-Purpose Numerical Simulator for Multiphase Fluid and Heat Flow*, Lawrence Berkeley National Laboratory, Berkeley, California.

Roylance, D., Yeh, A., and Lin, K. (2003), "Inverse Modelling of the Fusime Geothermal Well using ITOUGH2," Unpublished Report.

Sohn, A. (1995), "Parallel N-ary Speculative Computation of Simulated Annealing," *IEEE Trans. Parallel Distrib. Syst.*, 6, 997–1005.

van Genuchten, M. T. (1980), "A closed-form equation for predicting the hydraulic conductivity of unsaturated soils," *Soil Science Society of America Journal*, 44, 892–898.